Rao-Blackwellization of Generalized Accept-Reject Schemes

George Casella, Christian P. Robert and Martin T. Wells *

November 22, 2000

Abstract

This paper extends the accept-reject algorithm to allow the proposal distribution to change at each iteration. We first establish a necessary and sufficient condition for this generalized accept-reject algorithm to be valid, and then show how the Rao-Blackwellization of Casella and Robert (1996) can be extended to this setting. An important application of these results is to the perfect sampling technique of Fill (1998), which is a generalized accept-reject algorithm in disguise.

Keywords: Monte Carlo methods, Accept–Reject, stopping rule, recycling, uniform variable.

*George Casella is Professor and Chair, Department of Statistics, University of Florida. Email: casella@stat.ufl.edu. Both Prof. Casella and Prof. Wells are supported by National Science Foundation Grant DMS-9971586. Christian P. Robert is Head, Laboratoire de Statistique, CREST, Insee, Paris and Professor, Université Paris 9–Dauphine. This research was done while Prof. Robert was visiting the Department of Biometics, Cornell University, supported by National Science Foundation Grant DMS-9971586 and the EU TMR network ERB-FMRX-CT96-0095 on "Computational and statistical methods for the analysis of spatial data". Email: xian@ceremade.u-dauphine.fr. Martin T. Wells is Professor of Social Statistics, Cornell University, Ithaca NY 14851. Email: mtw1@cornell.edu.

1 Introduction

Accept-reject algorithms are based on the use of a proposal distribution g which serves to simulate from a given target density f, when the ratio f/g is bounded by $1/\epsilon$, say. The standard Accept-Reject Algorithm is

---

Algorithm $\mathbf{A_1}$ – Accept–Reject

`At iteration` i $(i \geq 1)$

1. `Generate` $X_i \sim g$ `and` $U_i \sim \mathcal{U}([0,1])$, independently.
2. `If` $U_i \leq \epsilon f(X_i)/g(X_i)$, `accept` $X_i \sim f$;
3. `otherwise, move to iteration` $i+1$.

Since the inequality is not always satisfied, the algorithm generates pairs (X_i, U_i) that are rejected. These pairs can be recycled in many ways, including the Rao-Blackwellizing approach by Casella and Robert (1996) which replaces the standard estimator δ based on the accepted pairs with the conditional expectation $\mathbb{E}[\delta|x_1, \ldots, x_n, n]$, which integrates out the uniform variables.

In a seemingly totally unrelated fashion, the development by Fill (1998) of an interruptible perfect sampling algorithm comes as an alternative to Propp and Wilson's (1996) coupling from the past technique, which allows the production of iid outputs from MCMC algorithms (see Dimakos (1999), Robert and Casella (1999) or Casella, Lavine and Robert (2000) for introductions to perfect sampling). At the core of Fill's algorithm, described in Section 2, is an accept-reject algorithm with the following specific features:

(a) the proposal distribution can be modified at each step by choosing a different number of forward iterations in the underlying MCMC algorithm,

(b) the acceptance probability and the proposal distribution are not known in closed form.

While this second difficulty is overcome by Fill's (1998) approach, the possibility of changing the proposal distribution at each failure/rejection (which is stressed by Fill as an important feature of his modification of the original coupling from the past approach to the point that Fill's algorithm is also called perfect sampling for the impatient user!) implies that his method does not fall in the category of a standard accept-reject algorithm and is thus not completely validated.

We give in this note a necessary and sufficient condition for this modified Accept–Reject algorithm to be valid and show that the Rao–Blackwellization technique of Casella and Robert (1996) also applies here, resulting in the use of the rejected samples to produce an improved estimator. Moreover, as the acceptance rate of the perfect sampling algorithms can be low, there is potential for a large amount of improvement.

2 Perfect Sampling

A *perfect sampling algorithm* for a Markov chain is an algorithm that produces a random variable that is exactly distributed according to the stationary distribution of the Markov chain. Moreover, the random variables are (typically) generated from the conditional distributions of the chain. Perfect sampling in Markov chains originated with the ingenious "coupling from the past" algorithm of Propp and Wilson (1996). In practice, however, this algorithm has some drawbacks, such as–for example–not being interruptible and thus creating biases in the output in cases of interruption for insufficient memory and such.

An alternative perfect sampling algorithm, based on rejection sampling, was proposed by Fill (1998). Since it is interruptible, Fill's perfect sampling algorithm seems to be somewhat more practical than coupling from the past, although it requires delicate reversibility and coupling arrangements as shown below.

Fill's algorithm can be described as follows:

(a) Starting at an arbitrary state 0, run a finite state Markov chain (X_i) for t (fixed) steps, and record $X_t = x$.

(b) Starting Markov chains in at all possible states at time t, run them in reversed time, coupled with the original chain.

(c) If all these chains have coalesced, that is, if they all are in state 0 at time 0, then accept $X_t = x$ as an observation from the stationary distribution. If not, reject X_t and start again, possibly with different values of 0 and of t.

The surprising feature with this method is that it is a rejection algorithm with the clever twist that the probability of acceptance is exactly the probability of coalescence. This circumvents the problem of calculating this acceptance probability, which is typically not feasible.

3 Generalized Accept–Reject Algorithms

3.1 Extension

We consider the following extension to the standard Accept–Reject algorithm:

Algorithm $\mathbf{A_2}$ – Generalized Accept–Reject

`At iteration` i $(i \geq 1)$

1. `Generate` $X_i \sim g_i$ `and` $U_i \sim \mathcal{U}([0,1])$, independently.
2. `If` $U_i \leq \epsilon_i f(X_i)/g_i(X_i)$, `accept` $X_i \sim f$;
3. `otherwise, move to iteration` $i+1$.

Thus, at each iteration i $(0 < i < \infty)$, the algorithm uses a different pair (g_i, ϵ_i) such that $\epsilon_i f(x)/g_i(x) \leq 1$,uniformly in x. Each of these pairs is thus acceptable for the original Accept–Reject scheme. However, the proposal distribution keeps changing at each reject iteration and may be more adaptive than the single Accept–Reject proposal distribution. For instance, if one region of the sample space does not lead to an acceptance after $1,000$ iterations, say, it is possible move to a completely different region and try again for a given number of iterations. Similarly, if the proposal distribution is parameterized by a parameter θ and if the current value of θ does not lead to a proposal, we can select a pre-determined sequence of values of θ to monitor the performance in simulating the distribution of interest f. The value of θ at the time of acceptance can then be exploited in further simulations without jeopardizing the independence properties of the algorithm.

In particular, since Fill's (1998) algorithm depends on a parameter t, which is the number of forward steps in the Markov chain and which can be modified at each iteration, by, for instance, doubling the value of t in a typical CFTP manner, it falls within this extension of the standard Accept–Reject algorithm.

3.2 A convergence result

A question of primary interest is obviously whether or not the extension of the Accept-Reject Algorithm remains valid. The answer is that it does not hold in full generality, in the sense that the distribution of the accepted random variable is not necessarily the correct one. A minimum requirement must be imposed on the sequence of the ϵ_i's (and hence on the g_i's).

If we denote by Z the random variable that is output by Algorithm A_2, Z has the cdf (in the univariate continuous case):

$$\begin{aligned}
P(Z \le z) &= \sum_{i=1}^{\infty} P(Z \le z, Z = X_i) \\
&= \sum_{i=1}^{\infty} P(X_i \le Z, U_i \le f(X_i)\epsilon_i/g_i(X_i)) \prod_{j=1}^{i-1} P(U_j \ge f(X_j)\epsilon_j/g_j(X_j)) \\
&= \sum_{i=1}^{\infty} \int_{-\infty}^{z} \frac{f(x)\epsilon_i}{g(x)} g(x)dx \prod_{j=1}^{i-1}(1-\epsilon_j) \\
&= \int_{-\infty}^{z} f(x)dx \sum_{i=1}^{\infty} \epsilon_i \prod_{j=1}^{i-1}(1-\epsilon_j)\,.
\end{aligned}$$

Therefore, the output is distributed from f if

$$\sum_{i=1}^{\infty} \epsilon_i \prod_{j=1}^{i-1}(1-\epsilon_j) = 1. \tag{1}$$

Theorem 3.1 *Equation (1) holds if, and only if, the series*

$$\sum_{i=1}^{\infty} \log(1-\epsilon_i) \tag{2}$$

diverges.

Proof. Note first that the left hand side of (1) necessarily converges to a limit less than, or equal to, 1 since

(a) for every $n \ge 1$,

$$\xi_n = \sum_{i=1}^{n} \epsilon_i \prod_{j=1}^{i-1}(1-\epsilon_j)$$

$$
\begin{aligned}
&= \epsilon_1 + (1-\epsilon_1)\{\epsilon_2 + (1-\epsilon_2)[\ldots(1-\epsilon_{n-1})\epsilon_n)\ldots]\} \\
&\le \epsilon_1 + (1-\epsilon_1)\{\epsilon_2 + (1-\epsilon_2)[\ldots\epsilon_{n-1} + (1-\epsilon_{n-1}))\ldots]\} \\
&= 1\,.
\end{aligned}
$$

(b) the sequence $\{\xi_n\}$ is increasing with n.

Now, $\{\xi_n\}$ converges to 1 if, and only if, for every $0 < \eta < 1$, there exists n_0 such that

$$\xi_n > 1 - \eta \qquad \text{for} \qquad n > n_0\,. \tag{3}$$

The condition (3) is equivalent to, for $n > n_0$,

$$
\begin{aligned}
& \epsilon_1 + (1-\epsilon_1)\{\epsilon_2 + (1-\epsilon_2)[\ldots(1-\epsilon_{n-1})\epsilon_n)\ldots]\} > 1-\eta \\
\Leftrightarrow\ & \epsilon_2 + (1-\epsilon_2)\{\epsilon_3 + \ldots(1-\epsilon_{n-1})\epsilon_n]\ldots\} > \frac{1-\epsilon_1-\eta}{1-\epsilon_1} = 1 - \frac{\eta}{1-\epsilon_1} \\
\Leftrightarrow\ & \ldots \\
\text{(4)}\quad \Leftrightarrow\ & \epsilon_n > 1 - \frac{\eta}{\prod_{i=1}^{n-1}(1-\epsilon_i)}\,.
\end{aligned}
$$

Now, the sequence $\omega_n = \prod_{i=1}^{n-1}(1-\epsilon_i)$ with $\omega_1 = 1$ is decreasing and nonnegative. Thus, it either converges to 0 or to $\alpha > 0$. If it converges to 0, that is, if (2) diverges, the ratio η/ω_n goes to $+\infty$ with n and the right hand side in (4) is negative for n large enough, which ensures that (3) holds.

If $\{\omega_n\}$ converges to $\alpha > 0$, the series (2) converges and $\log(1-\epsilon_n)$ goes to 0 as n goes to infinity by Cauchy's criterion. Thus, $\{\epsilon_n\}$ converges to 0. Therefore, for δ small enough, there exists n_1 such that $\epsilon_n < \delta$ for $n > n_1$. If one chooses η such that

$$1 - \frac{\eta}{\alpha} = \delta$$

and if (3) holds, one gets

$$\epsilon_n < \delta < \epsilon_n$$

for $n > \max(n_0, n_1)$, which is impossible. □□

3.3 Consequences

This result has several implications. First, it shows that continued modifications of the proposal distribution in the accept-reject algorithm are legitimate as long as the acceptance rate ϵ_n does not converge to zero too fast.

Also, contrary to intuition, it also states that the acceptance rate ϵ_n does not have to go to 1 with n. Therefore, some ϵ_n's (and even an infinity of them) may be equal to 0, and the algorithm remains valid. This may sound like a completely artificial property but there are settings, including Fill's (1998), where the bound $1/\epsilon$ is unknown and may be arbitrarily large.

Note also that, if one ϵ_n is equal to 1, the sequence terminates. Theorem 3.1 obviously applies to and validates the generalized accept-reject algorithm when ϵ_n is constant, but also when the ϵ_n's are periodic in n, and when the sequence $\{\epsilon_n\}$ is uniformly bounded away from 0.

Theorem 3.1 does not validate Fill's (1998) algorithm in all situations, since the output distribution is correct only if (1) holds. The difficulty then lies in establishing this equality without the ϵ_i's being available, which is the essence of Fill's technique. However, following the above remark, if the number of possible pairs $(t, 0)$ is finite and if the selection is periodic, Fill's algorithm is indeed valid, provided some ϵ_i's are different from 0.

4 Rao-Blackwellization

We can now extend Casella and Robert's (1996) improvement in this more general setup. The output from the generalized accept-reject algorithm is as follows: A sequence $Y_1, Y_2, \ldots$ of independent random variables is generated from the g_i's along with a corresponding sequence $U_1, U_2, \ldots$ of uniform random variables.

Given a function h, the Accept–Reject estimator of $\mathbb{E}^f\{h(X)\}$, based upon a sample $X_1, \ldots, X_t$ made of the t accepted values among the Y_j's, is given by

$$\hat{\tau}_1 = \frac{1}{t} \sum_{i=1}^{t} h(X_i)\,. \tag{5}$$

For a fixed sample size t, i.e. for a fixed number of accepted random variables, the number of generated Y_i's is a random integer N satisfying

$$\sum_{i=1}^{N} I(U_i \le w_i) = t \qquad \text{and} \qquad \sum_{i=1}^{N-1} I(U_i \le w_i) = t-1,$$

where we define $w_i = f(Y_i)\epsilon_i / g_i(Y_i)$. Since $\hat{\tau}_1$ can be written as

$$\hat{\tau}_1 = \frac{1}{t} \sum_{i=1}^{N} \mathbb{I}(U_i \le w_i) h(Y_i),$$

the conditional expectation

$$\hat{\tau}_2 = \frac{1}{t}\mathbb{E}\left\{\sum_{i=1}^{N} \mathbb{I}(U_i \le w_i) h(y_i) \Big| N, y_1, \ldots, y_N\right\} \tag{6}$$

improves upon (5).

The joint distribution of $(N, Y_1, \ldots, Y_N, U_1, \ldots, U_N)$ is given by

$$\begin{aligned} &P(N = n, Y_1 \le y_1, \ldots, Y_n \le y_n, U_1 \le u_1, \ldots, U_n \le u_n) = \\ &\int_{-\infty}^{y_n} g_n(t_n)(U_n \wedge w_n) dt_n \int_{-\infty}^{y_1} \cdots \int_{-\infty}^{y_{n-1}} g_1(t_1) \ldots g_{n-1}(t_{n-1}) \times \\ &\qquad \sum_{(i_1,\ldots,i_{t-1})} \prod_{j=1}^{t-1} (w_{i_j} \wedge U_{i_j}) \prod_{j=t}^{n-1} (U_{i_j} - w_{i_j})^+ dt_1 \ldots dt_{n-1}\,, \end{aligned}$$

where the last sum is over all subsets of $\{1, \ldots, n-1\}$ of size $t-1$. Therefore, the conditional density of the U_i's is given by

$$\begin{aligned} f(u_1, \ldots, u_n | N = n, y_1, \ldots, y_n) &= \left\{\sum_{(i_1,\ldots,i_{t-1})} \prod_{j=1}^{t-1} w_{i_j} \prod_{j=t}^{n-1} (1 - w_{i_j})\right\}^{-1} \\ \times \left\{\sum_{(i_1,\ldots,i_{t-1})} \prod_{j=1}^{t-1} \mathbb{I}(U_{i_j} \le w_{i_j}) \prod_{j=t}^{n-1} \mathbb{I}(U_{i_j} > w_{i_j})\right\} &\frac{\mathbb{I}(U_n \le w_n)}{w_n}\,. \end{aligned}$$

Using this distribution we can calculate, conditional on $(N, y_1, \ldots, y_N)$, the probability ρ_i of the events $\{U_i < w_i\}$ and thus derive the weights of $h(y_i)$ in the estimator $\hat{\tau}_2$. The calculations involve averaging over permutations of the realized sample and yield, for $i < n$,

$$\rho_i = w_i \sum_{(i_1,\ldots,i_{t-2})} \prod_{j=1}^{t-2} w_{i_j} \prod_{j=t-1}^{n-2} (1 - w_{i_j}) \Big/ \sum_{(i_1,\ldots,i_{t-1})} \prod_{j=1}^{t-1} w_{i_j} \prod_{j=t}^{n-1} (1 - w_{i_j})\,, \tag{7}$$

while $\rho_n = 1$. The numerator sum is over all subsets of $\{1, \ldots, i-1, i+1, \ldots, n-1\}$ of size $t-2$, and the denominator sum is over all subsets of size $t-1$. Everything being reproduced as in Casella and Robert (1996), except the definition of the w_i's which involves the pairs (g_i, ϵ_i). The following result holds with the proof identical to that in Casella and Robert (1996):

Theorem 4.1 *For $N = n$, the Rao-Blackwellized version of (5) is given by*

$$\hat{\tau}_2 = \frac{1}{t} \sum_{i=1}^{n} \rho_i h(Y_i)$$

where ρ_i is provided by equation (7).

In addition, the same recursion formula as in Casella and Robert (1996) applies, namely if we define

$$S_k(m) = \sum_{(i_1,\ldots,i_k)} \prod_{j=1}^{k} w_{i_j} \prod_{j=k+1}^{m} (1 - w_{i_j}), \qquad (k \le m < n)$$

with $\{i_1, \ldots, i_m\} = \{1, \ldots, m\}$, $S_k(m) = 0$ for $k > m$ and $S_k^i(i) = S_k(i-1)$, we can recursively calculate

$$\begin{aligned} S_k(m) &= w_m S_{k-1}(m-1) + (1 - w_m) S_k(m-1), \\ S_k^i(m) &= w_m S_{k-1}^i(m-1) + (1 - w_m) S_k^i(m-1), \qquad (m > i) \end{aligned}$$

and derive ρ_i as

$$\rho_i = w_i \; S_{t-2}^i(n-1)/S_{t-1}(n-1) \qquad (i < n)\,.$$

The application of this result to Fill's algorithm requires some further work since, in that case, the weights $w_i = f(x_i)/K^t(0, x_i)$ are not directly available. Note however that in some setups $K^t(0, x)$ may be known, while, in others, it can be estimated, since it is also equal to the probability of acceptance, that is, the probability of coalescence in state 0. Thus, we can implement the Rao-Blackwellized improvement with estimated weights.

References

Casella, G., Lavine, M., and Robert, C.P. (2000) An introduction to perfect sampling. Tech. report, Department of Biometrics, Cornell University.

Casella, G., and Robert, C.P. (1996) Rao-Blackewellization of sampling schemes. *Biometrika* **83**(1), 81–94.

Dimakos, X. (1999) A guide to perfect simulation. *Intern. Statist. Review.*

Fill, J.A. (1998) An interruptible algorithm for perfect sampling via Markov chains. *Annals of Applied Probability*, **8**(1):131–162.

Fill, J.A., Machida, M., Murdoch, D.J., and Rosenthal, J.S. (1999) Extension of Fill's perfect rejection sampling algorithm to general chains, 1999. Preprint, John Hopkins.

Fismen, M. (1998) Exact simulation using Markov chains. Technical Report 6/98, Institutt for Matematiske Fag, Olso. Diploma-thesis.

Møller, J. and Nicholls, G. K. (1999) Perfect simulation for sample-based inference. Preprint, Aalborg.

Propp, J.G. and Wilson, D.B. (1998) How to get a perfectly random sample from a generic Markov chain and generate a random spanning tree of a directed graph. *Journal of Algorithms*, **27**:170–217.

Propp, J.G. and Wilson, D.B. (1996) Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures and Algorithms*, **9**(1& 2):223–252.

Robert, C.P. and Casella, G. (1999) *Monte Carlo Statistical Methods.* Springer Verlag, New York.